AMENDMENTS TO THE CLAIMS

Claim 1 (cancelled).

Claim 2 (currently amended): The method as in claim 3 further including populating a destination image with extracted contents of the source disk in which the destination image has files, attributes, and structural relationships between files identical to [[file]] files associated with the source disk.

Claim 3 (previously presented):   A method for creating an image of a source disk of a first computer on a second computer that includes an operating system that has file system software, said method comprising:

loop-back mounting a simulated source disk in the second computer so that the simulated source disk is accessible by the operating system as a local disk; and

configuring the simulated source disk as a proxy for the source disk by intercepting sector-based I/O requests directed to the simulated source disk and retrieving source disk data from the source disk according to the intercepted sector-based I/O requests.

Claim 4 (previously presented):   The method as in claim 3, further comprising forwarding the intercepted sector-based I/O requests to the first computer over a network.

Claim 5 (previously presented):   The method as in claim 4, in which the source disk is associated with a first computer that has a memory, further comprising:

loading an imaging client program in the memory of the first computer, the imaging client program not being resident on the source disk; and

passing the intercepted sector-based I/O requests to the imaging client program, the imaging client program directing the intercepted sector-based I/O requests to the source disk.

Claim 6 (previously presented):   The method as in claim 5, further comprising:

loading a secondary operating system in the memory of the first computer, said secondary operating system not being present on the source disk and mediating I/O requests between the imaging client program and the source disk.

Claim 7 (previously presented):   The method as in claim 2 further comprising:

mounting the destination image in an uninitialized state in the second computer as a simulated destination disk;

intercepting sector-based I/O requests directed to the simulated destination disk and directing

the contents of the intercepted sector-based I/O requests to the destination image;

retrieving partition and file system layout information from the source disk;

formatting the simulated destination image to have the same partitioning and file system as the simulated source disk and thus of the source disk; and

copying files of at least one file system of the simulated source disk to the corresponding file system of the simulated destination disk.

Claim 8 (previously presented): The method as in claim 7, further comprising converting the intercepted sector-based I/O requests to the simulated destination disk into sector accesses within the destination image.

Claim 9 (previously presented): The method as in claim 7, in which the destination image is a virtual disk file associated with a virtual computer.

Claim 10 (previously presented): The method as in claim 9, in which the first computer is a physical computer and the source disk is a physical disk associated with the physical computer.

Claim 11 (previously presented): The method as in claim 9, in which the virtual disk file is a sparse virtual disk, having a predetermined capacity and initial sector contents with null values.

Claim 12 (previously presented): The method as in claim 7, in which the source disk is a source virtual disk.

Claim 13 (previously presented): The method as in claim 12, in which the destination disk is a physical disk.

Claim 14 (currently amended): The method as in claim 7, in which the source disk is a first virtual disk associated with a first virtual computer and the destination disk is a second virtual disk associated with a second virtual computer.[[.]]

Claim 15 (previously presented): The method as in claim 7, in which the first computer is the same as the second computer.

Claim 16 (previously presented): A method for creating an image of a source disk of a first computer, in which contents of the source disk are arranged according to at least one source file system, comprising:

in a second computer that includes an operating system that has file system software that detects a file system of disks mounted in the second computer, while the source disk is in an unmodified, unprepared state, extracting the contents of the source disk, defining extracted contents,

and populating a destination image with the extracted contents of the source disk such that the destination image may have a different sector-by-sector content than the source disk but a destination file system logically equivalent to the at least one source file system, with identical files, attributes, and structural relationships between files as the source disk;

mounting a simulated source disk in the second computer so that the simulated source disk is accessible by the operating system as a local disk;

configuring the simulated source disk as a proxy for the source disk by intercepting sector-based I/O requests directed to the simulated source disk and retrieving source disk data from the source disk according to the intercepted sector-based I/O requests;

forwarding the intercepted sector-based I/O requests to the first computer;

loading an imaging client program into a memory of the first computer;

passing the intercepted sector-based I/O requests to the imaging client program, the imaging client program directing the intercepted sector-based I/O requests to the source disk;

mediating, by the operating system, sector-based I/O requests between the imaging client and the source disk;

mounting the destination image in an uninitialized state in the second computer as a simulated destination disk;

intercepting sector-based I/O requests directed to the simulated destination disk and directing results of the intercepted sector-based I/O requests to the destination image;

converting the intercepted sector-based I/O requests to the simulated destination disk into sector accesses within the destination image;

retrieving partition and file system layout information from the source disk;

formatting the simulated destination image to have the same partitioning and file system(s) as the simulated source disk and thus of the source disk; and

copying files of at least one file system of the simulated source disk to the corresponding file system of the simulated destination disk.

Claim 17 (cancelled).

Claim 18 (currently amended):   A system for creating an image of a source disk in which contents of the source disk are arranged according to at least one source file system, said system comprising:

a first computer having the source disk; and

a second computer having a memory with an operating system and an imaging server residing therein, the imaging server including computer executable instructions having code to create a simulated source disk that is a representation of information stored on the source disk and is accessed by the operating system as a local disk; and code to mount the simulated source disk in the second computer, with said memory including file system drivers to detect a file system of the simulated source disk and a network ~~loopback~~ loop-back driver intercepting sector-based I/O requests directed to the simulated source disk and retrieving source disk data from the source disk according to intercepted sector-based I/O requests intercepted by the network loopback driver, defining intercepted sector based I/O requests.

Claim 19 (previously presented): The system as in claim 18, further comprising a network adapter, residing in said memory, to forward the intercepted sector-based I/O requests to the first computer.

Claim 20 (previously presented): The system as in claim 19, further comprising:

a first computer memory within the first computer; and

an imaging client installed in the first computer memory, said imaging client comprising computer-executable instructions that include code to receive any source disk I/O requests issued from the second computer to the first computer, code to direct the intercepted sector-based I/O requests to the source disk, and code to pass the retrieved source disk data to the second computer in response to the source disk I/O requests.

Claim 21 (currently amended): The system as in claim 18 wherein the imaging server further includes code to generate a simulated destination disk in response to the second computer mounting the destination image, with said memory further including a local ~~loopback~~ loop-back driver, a local adapter and a formatting module, with the local ~~loopback~~ loop-back driver intercepting sector-based I/O requests directed to the simulated destination disk and retrieving partition and file system layout information from the source disk, the local adapter comprising code to convert the intercepted sector-based I/O requests to the simulated destination disk into sector accesses within the destination image and the formatting module comprising code to format the destination image to have the same partitioning and file system(s) as the simulated source disk and thus of the source disk, the imaging server having code to copy files of at least one file system of the simulated source disk

to the corresponding file system of the simulated destination disk.

Claim 22 (previously presented):   The system as in claim 21, in which the source disk is a virtual disk.

Claim 23 (previously presented):   The system as in claim 22, in which the destination disk is a physical disk.

Claim 24 (previously presented):   The system as in claim 21, in which the destination image is a virtual disk file associated with a virtual computer.

Claim 25 (previously presented):   The system as in claim 24, in which the first computer is a physical computer and the source disk is a physical disk associated with the physical computer.

Claim 26 (cancelled)

Claim 27 (currently amended):   A system for creating an image of a source disk of a first computer, which has a memory and in which contents of the source disk are arranged according to at least one source file system, comprising:

a second computer;

a server operating system that resides in the second computer;

file system drivers within the server operating system of the second computer automatically detecting at least one file system of disks mounted in the second computer;

an imaging server running within the second computer and comprising computer-executable instructions:

for extracting the contents of the source disk, defining extracted contents, and populating a destination image with the extracted contents of the source disk such that the destination image may have a different sector-by-sector content than the source disk but a destination file system logically equivalent to the at least one source file system;

for creating a simulated source disk corresponding to the source disk;

while the source disk is in an unmodified, unprepared state, for mounting the simulated source disk in the second computer, file system drivers thereby automatically detecting [[the]] a file system of the simulated source disk and therefore of the source disk and exposing the file system to software running on the second computer;

a network loopback loop-back driver intercepting sector-based I/O requests directed to the simulated source disk;

a network adapter forwarding the intercepted sector-based I/O requests to the first computer;

an imaging client installed in the memory of the first computer, said imaging client comprising computer-executable instructions[[;]]:

for receiving any source disk I/O requests issued from the second computer to the first computer,

for directing the intercepted sector-based I/O requests to the source disk, and

for passing to the second computer source disk data retrieved in response to the source disk I/O requests;

a simulated destination disk generated by mounting the destination image in an uninitialized state in the second computer;

a local ~~loopback~~ <u>loop-back</u> driver intercepting sector-based I/O requests directed to the simulated destination disk and retrieving partition and file system layout information from the source disk;

a local adapter comprising computer-executable instructions for converting the intercepted sector-based I/O requests to the simulated destination disk into sector accesses within the destination image;

a formatting module comprising computer-executable instructions for formatting the destination image to have the same partitioning and file system as the simulated source disk and thus of the source disk; and

the imaging server further comprising computer-executable instructions for copying files of at least one file system of the simulated source disk to the corresponding file system of the simulated destination disk.

Claims 28-52 (cancelled).